MUSIC STORE DATABASE

(POSTGRE SQL PROJECT)

DONE BY – TEESHA SHARMA

SQL PROJECT- MUSIC STORE DATA ANALYSIS

Question Set 1 - Easy

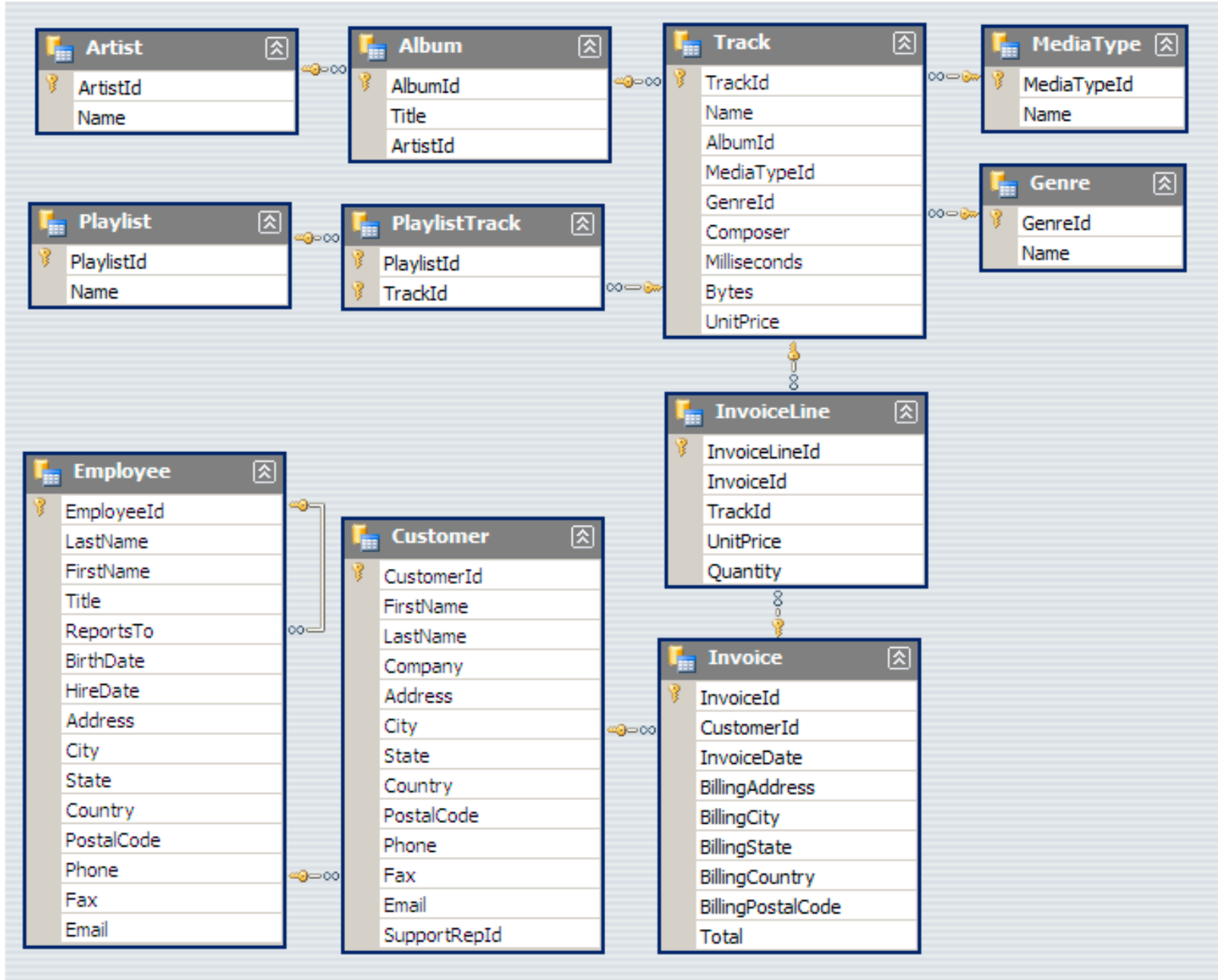
1. Who is the senior most employee based on job title?
2. Which countries have the most Invoices?
3. What are top 3 values of total invoice?
4. Which city has the best customers? We would like to throw a promotional Music Festival in the city we made the most money. Write a query that returns one city that has the highest sum of invoice totals. Return both the city name & sum of all invoice totals
5. Who is the best customer? The customer who has spent the most money will be declared the best customer. Write a query that returns the person who has spent the most money

Question Set 2 – Moderate

1. Write query to return the email, first name, last name, & Genre of all Rock Music listeners. Return your list ordered alphabetically by email starting with A
2. Let's invite the artists who have written the most rock music in our dataset. Write a query that returns the Artist name and total track count of the top 10 rock bands
3. Return all the track names that have a song length longer than the average song length. Return the Name and Milliseconds for each track. Order by the song length with the longest songs listed first

Question Set 3 – Advance

1. Find how much amount spent by each customer on artists? Write a query to return customer name, artist name and total spent
2. We want to find out the most popular music Genre for each country. We determine the most popular genre as the genre with the highest amount of purchases. Write a query that returns each country along with the top Genre. For countries where the maximum number of purchases is shared return all Genres
3. Write a query that determines the customer that has spent the most on music for each country. Write a query that returns the country along with the top customer and how much they spent. For countries where the top amount spent is shared, provide all customers who spent this amount


Artist
ArtistId
Name
Album
AlbumId
Title
ArtistId
Track
TrackId
Name
AlbumId
MediaTypeId
GenreId
Composer
Milliseconds
Bytes
UnitPrice
MediaType
MediaTypeId
Name
Genre
GenreId
Name
Playlist
PlaylistId
Name
PlaylistTrack
PlaylistId
TrackId
InvoiceLine
InvoiceLineId
InvoiceId
TrackId
UnitPrice
Quantity
Employee
EmployeeId
LastName
FirstName
Title
ReportsTo
BirthDate
HireDate
Address
City
State
Country
PostalCode
Phone
Fax
Email
Customer
CustomerId
FirstName
LastName
Company
Address
City
State
Country
PostalCode
Phone
Fax
Email
SupportRepId
Invoice
InvoiceId
CustomerId
InvoiceDate
BillingAddress
BillingCity
BillingState
BillingCountry
BillingPostalCode
Total

Solutions Set 1 - Easy

```sql
--Q1
select * from employee
order by levels desc
limit 1

--Q2
select billing_country, count(billing_country) from invoice
group by 1
order by 2 desc
limit 1
```

```sql
--Q3
select total from invoice
order by 1 desc
limit 3

--Q4
select billing_city,sum(total) from invoice
group by 1
order by 2 desc
limit 1
```

```sql
--Q5
select
c.customer_id,c.first_name, c.last_name,sum(i.total) from customer c
inner join invoice i
on c.customer_id = i.customer_id
group by 1
order by 4 desc
limit 1
```

Solutions Set 2 - Moderate

```
--MODERATE
--Q1

select  distinct c.email,c.first_name,c.last_name
from customer c
inner join invoice i
on c.customer_id = i.customer_id
inner join invoice_line il
on i.invoice_id = il.invoice_id
inner join track t
on il.track_id = t.track_id
where c.email like 'a%'
group by 1,3,2
order by 3
```

```
--Q2
select a.name,count(distinct t.track_id),
g.name from artist a
inner join album ab
on a.artist_id = ab.artist_id
inner join track t
on ab.album_id = t.album_id
inner join genre g
on t.genre_id = g.genre_id
where g.name like 'Rock'
group by 1,3
order by 2 desc
limit 10
```

```
--Q3
select name,milliseconds from track
where milliseconds>( select avg(milliseconds) from track)
group by 1,2
order by 2 desc;
```

Solutions Set 3 - Advance

```sql
--Q1
With best_selling_artist as (
        select a.artist_id,a.name, sum(il.unit_price*il.quantity)
        from invoice_line il
        inner join track t on il.track_id = t.track_id
        inner join album ab on t.album_id = ab.album_id
        inner join artist a on ab.artist_id = a.artist_id
        group by 1,2
        order by 3 desc
        limit 1 )

select c.first_name||' '||c.last_name as full_name,
a.name, sum(il.unit_price*il.quantity)
from customer c
inner join invoice i on c.customer_id = i.customer_id
inner join invoice_line il on i.invoice_id = il.invoice_id
inner join track t on il.track_id = t.track_id
inner join album ab on t.album_id = ab.album_id
inner join artist a on ab.artist_id = a.artist_id
inner join best_selling_artist bsc on bsc.artist_id = ab.artist_id
group by 1,2
order by 3 desc;
```

```sql
--Q2
with popular_genre as (
select c.country,il.quantity,g.name,
row_number() over(partition by c.country order by count(il.quantity) desc) as row_no
from customer c
inner join invoice i on  c.customer_id = i.customer_id
inner join invoice_line il on i.invoice_id = il.invoice_id
inner join track t on il.track_id = t.track_id
inner join genre g on t.genre_id = g.genre_id
group by 1,2,3
order by 1 desc,2 asc)
select * from popular_genre where row_no <= 1
```

```sql
--Q3
with abc as(select i.billing_country, c.customer_id,
c.first_name||' '||c.last_name as full_name,
sum(i.total),
row_number() over(partition by c.country order by sum(i.total) desc) as row_no
from invoice i
inner join customer c on i.customer_id = c.customer_id
inner join invoice_line il on i.invoice_id = il.invoice_id
inner join track t on il.track_id = t.track_id
group by 1,2,3
order by 1 asc,2 asc)
select * from abc where row_no <= 1
```